

Claims

1. A computer system (999) for handling incremental data comprising:

5 a server-controller (101-1) for receiving a modification-request from a client (900) to modify an original model (200-T1) of an application component that is stored on the server (901) into a modified model (200-T2) of the application component;

10 a server-renderer (101-2) for generating at least one browser-increment (300-I) that corresponds to the difference between the original model (200-T1) and the modified model (200-T2);

15 a client-assembler (100-1) receiving the at least one browser-increment (300-I) from the server (901) and updating at the client (900) an original DOM component (300-T1) of a browser component with the at least one browser-increment (300-I), resulting in a modified document object model (DOM) component (300-T2) that corresponds to the modified model (200-T2), wherein the original DOM component (300-T1) corresponds to the original model (200-T1); and

20 25 a client-controller (100-2) for generating the modification-request.

2. The computer system (999) of claim 1, wherein the client-controller (100-2) stores the at least one browser-increment (300-I) in a cache-memory (920-C) of the client (900) and instructs the client-assembler (100-1) to deactivate the at least one browser-increment (300-I) upon receiving a deactivation-request (DAR) (DAR).

5
10 3. The computer system (999) of claim 2, wherein the client-controller (100-2) retrieves the at least one browser-increment (300-I) from the cache-memory (920-C) and instructs the client-assembler (100-1) to reactivate the at least one browser-increment (300-I) upon receiving a reactivation-request (RAR).

15
20 4. The computer system (999) according to any of the claims 1 to 3, wherein the client-controller (100-2) instructs the client-assembler (100-1) to reset the original or modified DOM component (300-T1, 300-T2) upon receiving a reset-request.

25 5. The computer system (999) according to any of the claims 1 to 4, wherein the original model (200-T1) and the modified model (200-T2) are defined by a component class selected from the group of Java class, Java Server Pages class, servlet class, Pascal class, C class, C++ class, and Business Server Pages class.

6. The computer system (999) according to any of the claims 1 to 5, wherein the browser component is defined by a component script class selected from the group of JavaScript class, JavaApplets class and VisualBasic Script class.

5

7. The computer system (999) of claim 5, wherein the component class implements at least a portion of the server-controller (101-1) and the server-renderer (101-2).

10

8. The computer system (999) of claim 6, wherein the component script class implements at least a portion of the client-controller (100-2) and the client-assembler (100-2).

15

9. The computer system (999) of claim 6, wherein the component script class and the component class have identical hierarchies.

10. A server (900) in a ³⁴computer system (999) for handling incremental data comprising:

5 a server-controller (101-1) for receiving a modification-request from a client-controller (100-2) of a client (900) in the computer system (999) to modify an original model (200-T1) of an application component that is stored on the server (901) into a modified model (200-T2) of the application component;

10 and

a server-renderer (101-2) for generating at least one browser-increment (300-I) that corresponds to the difference between the original model (200-T1) and the modified model (200-T2); the at least one browser-increment (300-I) made to be sent to a client-assembler (100-1) of the client (900) for updating an original DOM component (300-T1) that corresponds to the original model (200-T1) with the at least one browser-increment (300-I), resulting in a modified DOM component (300-T2) that corresponds to the modified model (200-T2).

35

11. A client (900) in a computer system (999) for handling incremental data comprising:
5 a client-controller (100-2) sending a modification-request to a server-controller (101-1) of a server (901) in the computer system (999); and
a client-assembler (100-1) receiving at least one browser-increment (300-I) from the server (901) and updating an original DOM component (300-T1)
10 that corresponds to an original model (200-T1) of an application component with the at least one browser-increment (300-I), resulting in a modified DOM component (300-T2) that corresponds to a modified model (200-T2) of the application component, wherein the server-controller (101-1) modifies the original model (200-T1) being stored on the server (901) into the modified model (200-T2); and a server-renderer (101-2) of the server (901) generates
15 the at least one browser-increment (300-I) that corresponds to the difference between the original model (200-T1) and the modified model (200-T2).
20

12. The client (900) of claim 11, wherein the client-controller (100-2) stores the at least one browser-increment (300-I) in a cache-memory (920-C) of the client (900) and instructs the client-assembler (100-1) to deactivate the browser-increment (300-I) upon receiving a deactivation-request (DAR).

13. The client (900) of claim 12, wherein the client-controller (100-2) retrieves the at least one browser-increment (300-I) from the cache-memory (920-C) and instructs the client-assembler (100-1) to reactivate the at least one browser-increment (300-I) upon receiving a reactivation-request (RAR).
5
14. The client (900) according to any of the claims 11 to 13, wherein the client-controller (100-2) instructs the client-assembler (100-1) to reset the original DOM component (300-T1) upon receiving a reset-request.
10

15. A method (400) for handling incremental data on a server (901) of a computer system (999) comprising the steps:

receiving (410) by a server-controller (101-1) a modification-request from a client-controller (100-2) belonging to a client (900) of the computer system (999) to modify an original model (200-T1) of an application component that is stored on the server (901) into a modified model (200-T2) of the application component;

5 generating (420) by a server-renderer (101-2) at least one browser-increment (300-I) that corresponds to the difference between the original model (200-T1) and the modified model (200-T2); and

10 sending (430) the at least one browser-increment (300-I) to a client-assembler (100-1) of the client (900) for updating on the client (900) an original DOM component (300-T1) that corresponds to the original model (200-T1) with the at least one browser-increment (300-I), resulting in a modified DOM component (300-T2) that corresponds to the modified model (200-T2).

16. A method (500) for handling incremental data on a client (900) of a computer system (999) comprising the steps:

5 sending (510) from a client-controller (100-2) a modification-request to a server-controller (101-1) of a server (901) of the computer system (999); and

10 receiving (520) by a client-assembler (100-1) at least one browser-increment (300-I) from the server (901) as a response to the modification request; and

15 updating (530) an original DOM component (300-T1) that corresponds to an original model (200-T1) of an application component with the at least one browser-increment (300-I), resulting in a modified DOM component (300-T2) that corresponds to a modified model (200-T2) of the application component, wherein the server-controller (101-1) modifies the original model (200-T1) being stored on the server (901) into the modified model (200-T2); and a server-renderer (101-2) of the server (901) generates the at least one browser-increment (300-I) that corresponds to the difference between the original model (200-T1) and the modified model (200-T2).

20 17. The method (500) of claim 16, comprising the further step:

25 30 storing (540) the at least one browser-increment (300-I) in a cache-memory (920-C) of the client (900).

35 18. The method (500) of claim 17, comprising the further step:

deactivating (550) the ³⁹ browser-increment (300-I) by the client-assembler (100-1) upon the client-controller (100-2) having received a deactivation-request (DAR).

5

19. The method (500) of claim 18, comprising the further steps:

retrieving (560) the at least one browser-increment (300-I) from the cache-memory (920-C); and

10 reactivating (570) the browser-increment (300-I) by the client-assembler (100-1) upon the client-controller (100-2) having received a reactivation-request (RAR).

15 20. A computer program product (101) comprising instructions that, when loaded into a memory (921) of a server (901), cause at least one processor (911) of the server (901) to execute the steps of claim 15.

20

21. A computer program product (100) comprising instructions that, when loaded into a memory (920) of a client (900), cause at least one processor (910) of the server (900) to execute the steps of any of the claims 16 to 19.

25

22. A computer system (999)⁴⁰ for handling incremental data comprising:

a client-controller (100-2) generating a modification-request;

5 a server-controller (101-1) modifying (703) a model (200-Tn) of an application component on a server (901) as a response to the modification-request;

10 a server-renderer (101-2) generating (801) at least one browser-increment (300-I) after the model (200-Tn) has been modified (703); and

15 a client-assembler (100-1) receiving the at least one browser-increment (300-I) from the server (901) and updating an instance of a browser component at the client (900) with the at least one browser-increment (300-I), wherein the browser component corresponds to the application component.